5054–5067 *Nucleic Acids Research, 2016, Vol. 44, No. 11* *Published online 17 May 2016*
doi: 10.1093/nar/gkw448

RNAontheBENCH: computational and empirical resources for benchmarking RNAseq quantification and differential expression methods

Pierre-Luc Germain[1], Alessandro Vitriolo[1,2], Antonio Adamo[1], Pasquale Laise[1], Vivek Das[1,2] and Giuseppe Testa[1,2,*]

[1]European Institute of Oncology, Department of Experimental Oncology, Via Adamello 16, 20139 Milano, Italy and [2]University of Milan, Department of Oncology and Hemato-Oncology, Via Festa del Perdono 7, 20122 Milano, Italy

Received January 18, 2016; Revised May 07, 2016; Accepted May 09, 2016

ABSTRACT

RNA sequencing (RNAseq) has become the method of choice for transcriptome analysis, yet no consensus exists as to the most appropriate pipeline for its analysis, with current benchmarks suffering important limitations. Here, we address these challenges through a rich benchmarking resource harnessing (i) two RNAseq datasets including ERCC ExFold spike-ins; (ii) Nanostring measurements of a panel of 150 genes on the same samples; (iii) a set of internal, genetically-determined controls; (iv) a reanalysis of the SEQC dataset; and (v) a focus on relative quantification (i.e. across-samples). We use this resource to compare different approaches to each step of RNAseq analysis, from alignment to differential expression testing. We show that methods providing the best absolute quantification do not necessarily provide good relative quantification across samples, that count-based methods are superior for gene-level relative quantification, and that the new generation of pseudo-alignment-based software performs as well as established methods, at a fraction of the computing time. We also assess the impact of library type and size on quantification and differential expression analysis. Finally, we have created a R package and a web platform to enable the simple and streamlined application of this resource to the benchmarking of future methods.

INTRODUCTION

RNA sequencing (RNAseq) has become the method of choice for transcriptome analysis, and a large variety of softwares and methods have been developed for the different steps leading to quantification and differential expression analysis (DEA). Several benchmarking efforts (1–12) have allowed their comparison, showing a fairly high reproducibility of the most popular methods (9), while highlighting their respective strengths and weaknesses (see especially (1) for alignment and (3) for differential expression). In addition, some studies demonstrated the effect of ambiguous alignments or multi-mapping reads on quantification biases (7,11), and of differential isoform length on gene-level quantification (4,13), although the effect of the latter in real datasets appears relatively small, and gene-level estimates were shown to be more stable than transcript-level ones (14). Regarding DEA, in addition to a comparison of statistical models (3), previous contributions have for instance studied the effect of ‘zero-count genes’ on some methods (6,15), and shown that isoform filtering can improve differential transcript usage (16).

While these efforts have all been critical to guide the choice and development of analysis methods, little consensus has yet emerged as to the most appropriate pipeline. Part of the problem is due to the constant release of new methods, which makes previous comparisons obsolete, and this is especially true with the recent emergence of new, alignment-free (or pseudo-alignment-based) methods (17). In addition, previous benchmarking efforts also have important internal limitations. The most important (and largely inevitable) is arguably the absence of a clear and robust gold standard on which to judge performance. In facing this issue, a first strategy has been to rely on simulated data that are generated however through in-built assumptions and might thus not accurately reflect the biological conditions, as well as the technical biases and variations of a real experiment. A second strategy has been to rely on established technologies such as microarrays or RT-qPCR, which have however a very different dynamic range and suffer themselves from systematic biases (18,19). Moreover, for practical reasons, RT-qPCR validation is typically performed on

[*]To whom correspondence should be addressed. Tel: +39 2 94375 105; Fax: +39 2 94375 990; Email: giuseppe.testa@ieo.eu
Present address: Antonio Adamo, Biological and Environmental Sciences and Engineering Division, King Abdullah University of Science and Technology, Thuwal, 23955-6900, Kingdom of Saudi Arabia.

a small set of genes, and/or for very few samples. Finally, another strategy has been to seek the method that maximizes the agreement between replicates, which guarantees some internal validity but cannot assess the accuracy of the quantification.

In addition, and despite the fact that a considerable proportion of RNAseq experiments are aimed at detecting differential expression, most benchmarking efforts have focused on absolute rather than relative (i.e. cross-sample) quantification, or have reduced the problem of relative quantification to that of differential expression analysis. Accurately quantifying differences across samples is necessary and prior not only to DEA, but also to a variety of other applications, such as classification and the reverse-engineering of gene regulatory networks. Moreover, the assessment of accuracy is generally performed by correlating observed expression values with expected ones, which can be heavily biased for heteroscedastic data that are non-homogeneously distributed across a large dynamic range (as is the case for RNAseq), and might thereby obfuscate proportionally small but potentially critical inaccuracies in the estimation of expression differences across samples.

To face these challenges, it is critical to use a plurality of complementary benchmarks relying on both empirical and *in silico* data, in order to mitigate their respective limitations. Therefore, we present here an empirically-grounded computational resource for benchmarking RNAseq analysis methods that addresses the main aforementioned issues: RNA on the Benchhmark of Expression by nCounter Hybridisation (RNAontheBENCH). RNAontheBENCH harnesses a number of innovative features: (i) a RNAseq dataset from 12 human induced pluripotent stem cell (iPSC) lines all including the External RNA Controls Consortium (ERCC) ExFold spike-ins (20), i.e. 92 transcripts in known concentrations, some of which are differentially-expressed across mixes; (ii) a panel of 150 genes representative of the transcriptome (Supplementary Figure S1), including multiple isoforms for some genes, measured in the same samples by the highly reproducible Nanostring nCounter technology (21); (iii) a set of genetically-determined, internal controls, i.e. genes present at different copy-numbers and known to be expressed linearly with copy-numbers; (iv) a further validation dataset for which the same RNA extraction was assayed using both RNAseq and Nanostring; (v) reanalyses of the Sequencing Quality Control (SEQC) dataset (9); (vi) accuracy metrics based on relative expression across samples, i.e. the comparison of feature-wise *z*-scores or foldchanges across samples; and (vii) an *in silico* dataset specifically designed to further establish the accuracy of relative expression at the transcript level. We use this resource to compare different approaches to each step of RNAseq analysis, from alignment to differential expression testing, and to assess the impact of coverage and library design.

The tested software can be divided into different categories on the basis of the task performed. First are genome alignment methods, which map genes onto genomic location but do not perform quantification. Of these, we tested Tophat 2.0.14 (22), STAR 2.4.1 (23), HISAT 0.1.6 (24) and MapSplice 2.1.9 (25). Second are methods quantifying genomic features on the basis of these previously aligned reads. The simplest such methods are based on counting the number of reads/fragment unambiguously overlapping a single feature (e.g. exon/transcript/gene), henceforth referred to as count-based methods. We chose not to test specifically the popular HTseq-count method (26), because featureCounts 1.4.4 proved equivalent and considerably faster (27). More refined quantification methods instead assign reads to features probabilistically, and among these we tested the popular Cufflinks 2.2.1 (22). Where possible, we tested both these methods (i.e. featureCounts and Cufflinks) in combination with each aligner. In addition, we tested RSEM 1.2.22 (28), which first performs alignment to the transcriptome using bowtie 2.1.0 (29) before performing transcript quantification through expectation-maximization. Third, a new and very different type of software skips traditional alignment and performs quantification at unprecedented speed through pseudo-alignment, and as such promises to transform RNAseq analysis. Of these, we tested Sailfish 0.7.6 (17), Salmon 0.5.0 (unpublished; manuscript: Patro, Duggal, and Kingsford, bioRxiv 2015, doi:10.1101/021592), and Kallisto 0.42.3 (unpublished but available at http://pachterlab.github.io/kallisto). Fourth and last are methods performing differential expression analysis on the basis of previously established quantifications. We tested DESeq2 1.10 (30,31), limma/voom 3.22.7 (32,33), edgeR 3.8.6 (34), sleuth 0.28.0 (unpublished but available at http://pachterlab.github.io/sleuth/) and Cuffdiff 2.2.1 (22). The software details are listed in Supplementary Table S1, while the different pipelines and parameters used are listed along with the results in Supplementary Table S2.

Rather than attempting an exhaustive review of all available methods, we focused on developing a resource that users could easily apply to other methods. To this end, all the benchmarking code is available through the RNAontheBENCH R package (source available on github at https://github.com/plger/RNAontheBENCH). In addition, and similarly to some previous efforts with microarrays (35), we developed a user-friendly online platform allowing an in-depth benchmarking of analysis methods (https://bio.ieo.eu/rnaseqBenchmark), making the current study easily accessible by the community and extensible to future software/methods.

MATERIALS AND METHODS

RNAseq and spike-in distribution

The RNAseq data has been previously published in GEO series GSE63055 (see Supplementary Table S3). The two different ExFold ERCC mixes were added to total RNA before RiboZero treatment and library preparation. For the main dataset, libraries were sequenced at a coverage of ∼45–75 M of 100 bp read pairs. For the smaller, validation dataset, libraries were sequenced at a coverage of ∼44–57 M of 51 bp read pairs. Reads were aligned to the NCBI GRCh38 genome including the spike-in and EBV sequences.

Nanostring panel

We designed a panel of Nanostring probes for 150 genes (including probes for different isoforms of two genes), and quantified them across all 12 samples (see Supplementary Table S4 for details on the probes' design). The genes include 21 genes that have different copy-numbers across samples (and were shown to be expressed accordingly), as well as other genes of interests and putatively differentially-expressed genes. Importantly, the panel of genes is representative of the whole human transcriptome in terms of transcript lengths, number of exons, and expression levels (Supplementary Figure S1A–C). Probes expected (by the Nanostring CodeSet design team) to cross-hybyidize with other genes were excluded from the analysis, as well as, for the primary dataset, one probe related to a gene which we had independent evidence to be differentially-expressed between the RNA extractions used respectively for RNAseq and Nanostring (see Supplementary Figure S2). For the purpose of comparing RNAseq and Nanostring quantifications, we summed the TPM values of transcripts matching each Nanostring probe. Since the Nanostring itself relies on housekeeping genes for normalization (GAPDH, TUBB, and TBP), for the purpose of comparison with Nanostring we re-normalized the RNAseq data using the same housekeeping genes (using geometric normalization of TPM values) when dealing with gene-level quantification. For transcript-level quantification, for many samples count-based methods were unable to assign reads unambiguously to the housekeeping transcripts, preventing their use for normalization. For this reason, we used TMM normalization (36) on final values for transcript-level analysis.

For differential-expression analysis, we compared the WBS samples to the 7dup samples (see Supplementary Table S3), and we considered a Nanostring probe to be differentially-expressed if a *t*-test on the log-transformed Nanostring intensities gave a *P*-value smaller than 0.01.

In silico data

We simulated 100 bp paired RNAseq reads from 17817 Refseq transcripts for eight samples (>5 M reads per sample) using the R package Polyester (37). For half of the samples, we introduced foldchanges ranging from 1.1 to 10-fold difference (random noise was added for half of the foldchanges). The exact foldchanges introduced are available in Supplementary Table S5, and the generated reads are available from our website (see below). The reads were simulated on the basis of Refseq transcripts with the simulate_experiment function, ran with the following parameters: (i) empirical fragment length distribution (distr = 'empirical'), (ii) Illumina sequencing error distribution (error_model = 'illumina5'), (iii) positional bias (bias = 'rnaf'), and (iv) read counts based on length (meanmodel = TRUE). Random quality scores following a usual Truseq pattern were added when converting reads to FASTQ.

SEQC data

We downloaded all SEQC (9) Illumina reads from the BGI site, merged the different lanes for each sample, aligned with HISAT and quantified with Cufflinks (as described in the HISAT-Cufflinks pipeline in Supplementary Table S2). Between-group DEA was performed using respectively all five replicates or replicates 1, 3, 5 of each group, while within-group DEA was performed comparing replicates 1, 2, 3 to replicates 4, 5 in each group.

Spike-in normalization

To avoid eventual biases in the loading of the spike-in mixes, we normalized the spike-ins independently of the rest of the transcriptome. However, most normalization methods assume that most of the genes/transcripts are *not* differentially-expressed, while most spike-ins are in fact in different concentration across mixes, biasing normalization. To address this issue, we homogenized the two mixes for the purpose of calculating normalization factors, by multiplying values of samples containing one of the mix by their expected foldchange to the other mix (the exact code can be found in the RNAontheBENCH R package). For the purpose of comparing quantification, we used linear normalization factors because they maximized the correlation. For the purpose of differential expression analysis, we used TMM normalization. Since Cuffdiff was not amenable to this spike-in homogenization and does not offer the freedom to input custom normalization factors, we hard-coded them into the program's source and recompiled a custom version to ensure comparability.

FPKM and TPM calculations

Unless specified otherwise, all tests were performed using normalized Transcripts Per Million (TPM). If the software produced TPM values, we used these directly. If it produced Fragments Per Kilobase of transcript per Million reads sequenced (FPKM), we converted these values to TPM. If it produced counts, we first calculated FPKM values (using effective length), and converted them to TPM.

For gene-level analysis, when the quantification method provided gene-level quantification, these were used (and eventually converted as described above); otherwise we summed the TPM values of the gene's transcripts.

Downsampling

For Salmon, the downsampling analysis was performed using the first *N* reads of the fastq files, where *N* was the number of reads corresponding to the given proportion for the given sample. For alignment-based methods, we instead used samtools 1.2 (38) to randomly select the corresponding number of reads from the aligned ones.

Software selection

The packages/methods used for comparison were selected either because (i) they are widely used, (ii) previous benchmarks have found them among the most accurate, and/or (iii) they were recently released and promising. The software versions are available in Supplementary Table S1. We used default or recommended settings unless we had reasons to think that some parameters would yield better results (the detailed parameters used can be found in Supplementary Table S2).

RESULTS

Transcripts and spike-ins detection

We first assessed the capacity of the different pipelines to detect the spike-ins as well as transcripts known to be expressed on the basis of the Nanostring data. We considered a transcript detected if the software gave it an expression level above zero. As shown in Figure 1A, the most important difference was the inability of count-based methods to detect a large proportion of transcripts. This is not surprising given that these methods normally discard most of the reads as ambiguous and hence require a greater coverage to quantify overlapping transcripts. Counts based on the MapSplice2 alignment showed the best detection rate (73%), followed by STAR (72%), but no difference in detection rate could be found among the other pipelines.

Instead the spike-ins, which were designed partly to assess the sensitivity of an experiment and therefore contain several RNAs in very low concentrations, did not show different detection rates between count-based quantification and other methods (Figure 1B). Since the spike-ins have no overlapping isoforms, this is consistent with the hypothesis of ambiguous reads being responsible for the count-based methods' poor performance in detecting transcripts. Finally, the spike-ins revealed a slightly better detection rate for Sailfish (86%, versus 84–85% for most others).

Spike-in and gene-level abundance estimates

To assess the accuracy of each pipeline in measuring spike-in and gene abundances, we calculated the overall correlations (using both Pearson and Spearman coefficients) of the quantification with, respectively, the real concentrations of the spike-ins (Figure 1C) and the Nanostring quantification of the genes (Figure 1D). In general, count-based methods performed worst, except for Sailfish which performed poorly *specifically* on the spike-in quantification. Salmon performed best for spike-ins, followed by Kallisto and RSEM; for gene abundance estimation, RSEM performed best but was nearly equalled by Cufflinks and pseudo-alignment-based methods. We also looked at the distribution of per-sample correlation, which displayed the same general trend but revealed a high variability of Cufflinks and Sailfish's spike-in (but not gene) quantification (Supplementary Figure S2).

Transcript abundance estimates

When performing the same analysis at the transcript level, count-based quantification performed extremely poorly, partly but not entirely due to its low detection rate (Supplementary Figure S3A). In addition, the abundance of a large number of transcripts was underestimated. An important reason is that when calculating expression in Fragments Per Kilobasepair of transcript per Million reads (FPKM), the entire length of the transcript (or minus the fragment length) is used, whereas large proportions of the transcript might not be uniquely mappable (e.g. exons shared by all isoforms of a gene) and hence have a null read count. If the length of these regions is counted in the transcript length, FPKM values (and derived Transcripts Per Million reads, TPM) will necessarily deviate from expected abundances. Therefore, for count-based methods we calculated FPKM not based on the real transcript length, but on its 'unique length', i.e. the length that is uniquely attributable to that transcript. While this represents only an approximation of the length across which fragments can be unambiguously assigned, it dramatically improved the performance of count-based methods (Supplementary Figure S3B). Nevertheless, these methods could not compare with alternative methods (Figure 1E-F), even when excluding undetected transcripts, and should therefore be avoided for (absolute) transcript abundance estimation. Among the remaining methods, Cufflinks and RSEM performed better than alignment-free methods. (See Supplementary Figure S4 for the distributions of per-sample correlations).

Accuracy of relative expression estimates

For many purposes such as differential expression analysis, it is much more important to be able to detect and quantify differences across samples than to be able to accurately quantify the relative abundance of each transcript within a sample. Given imperfect quantification estimates, the methods that perform best at one task do not necessarily perform best at the other, and we therefore tested the accuracy of quantification of differences across samples.

In our experimental design, each sample was spiked-in with one of two mixes of the ERCC ExFold kit, each containing different concentrations of the same set of spike-ins (see Supplementary Table S2 for the mix distribution across samples). To assess the accuracy of relative spike-in abundance estimates, we therefore simply correlated the measured and the expected foldchanges between mixes.

For gene-level and transcript-level differences, we calculated both the foldchange-to-the-mean and the z-score for each gene/transcript. However, given that the foldchange-to-the-mean is easily distorted (e.g. random differences at low expression levels), we focused on the *z*-scores, which provide more stable measure of relevant differences by scaling deviations from the mean on the standard deviation.

Importantly, when quantifying differences at the gene level, count-based methods performed better than other methods (Figure 2A), confirming the hypothesis that the ability to quantify differential expression does not necessarily hinge on the ability to quantify abundance, and warranting the use of count-based methods for gene-level differential-expression analysis. These results were partly confirmed when looking at the foldchange between spike-in mixes (Figure 2B), although the differences are small in magnitude (foldchange correlation ranging from 0.789 to 0.82): with the exception of STAR-based counts, count-based methods had a higher correlation with real foldchange, and a lower median or total error (Supplementary Table S2). Pseudo-alignment methods were in the middle, while Cufflinks (with the exception of the Tophat-Cufflinks pipeline) and RSEM performed worst.

Given the small magnitude of these differences, we sought to further validate them using genetically-determined internal controls: genes of the Williams-Beuren Syndrome Critical Region (WBSCR), which are either duplicated or hemizygously deleted in, respectively, Somerville-van der

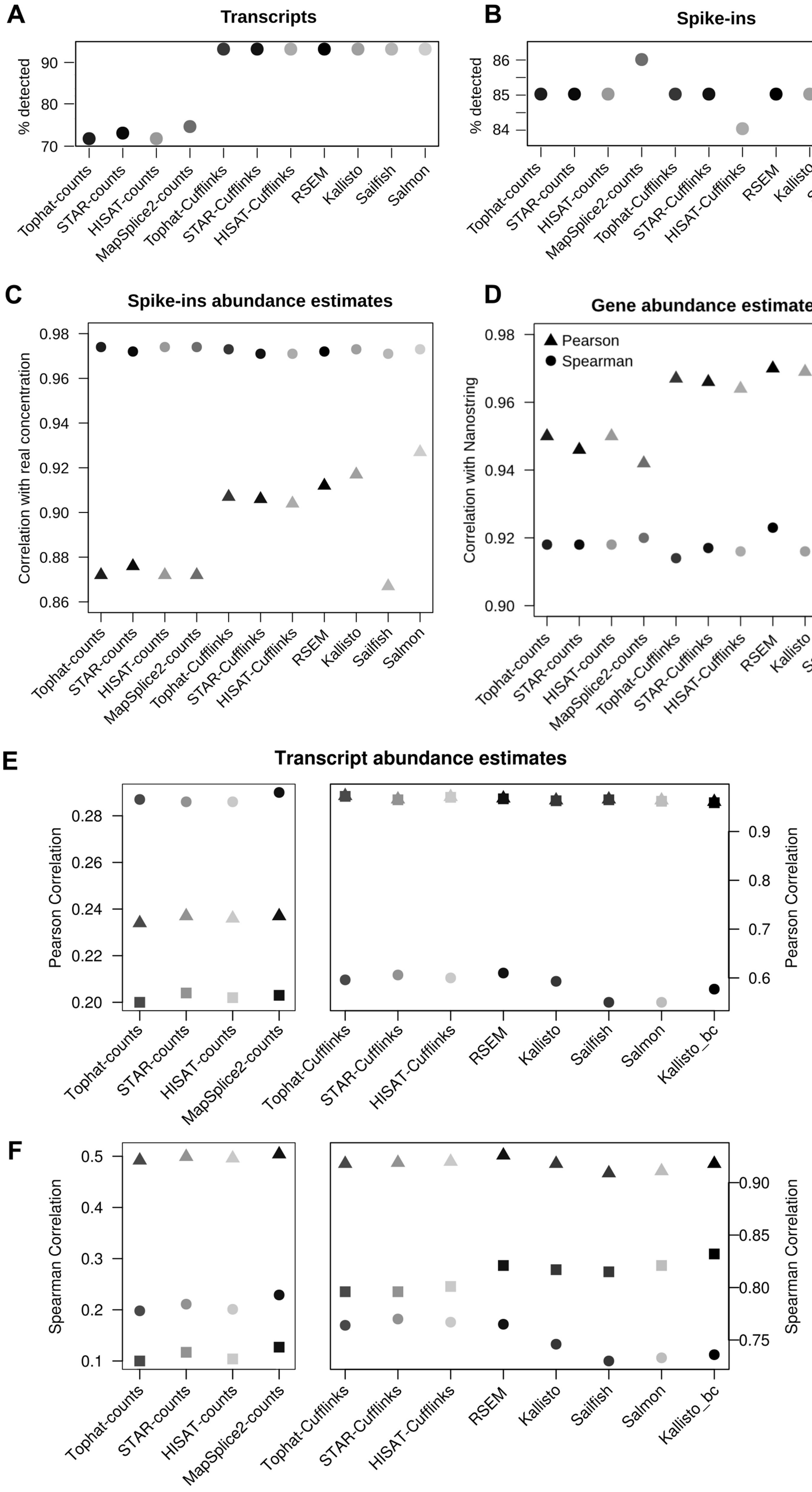


Figure 1. Accuracy of abundance estimates. (**A**) Proportion of transcripts detected by each pipeline. (**B**) Proportion of spike-ins detected. (**C**) Correlation of spike-in abundance estimates with real spike-in concentrations. Triangles indicate Pearson's correlation, while discs indicate the Spearman's correlation. (**D**) Correlation of gene expression estimates with Nanostring. Triangles indicate Pearson's correlation, while discs indicate the Spearman's correlation. Pearson (**E**) or Spearman (**F**) correlation of transcript abundance estimates with Nanostring. Triangles indicate the overall correlation, discs indicate the correlation for transcripts expressed below the median expression level, and squares indicate the correlation for transcript above the median.

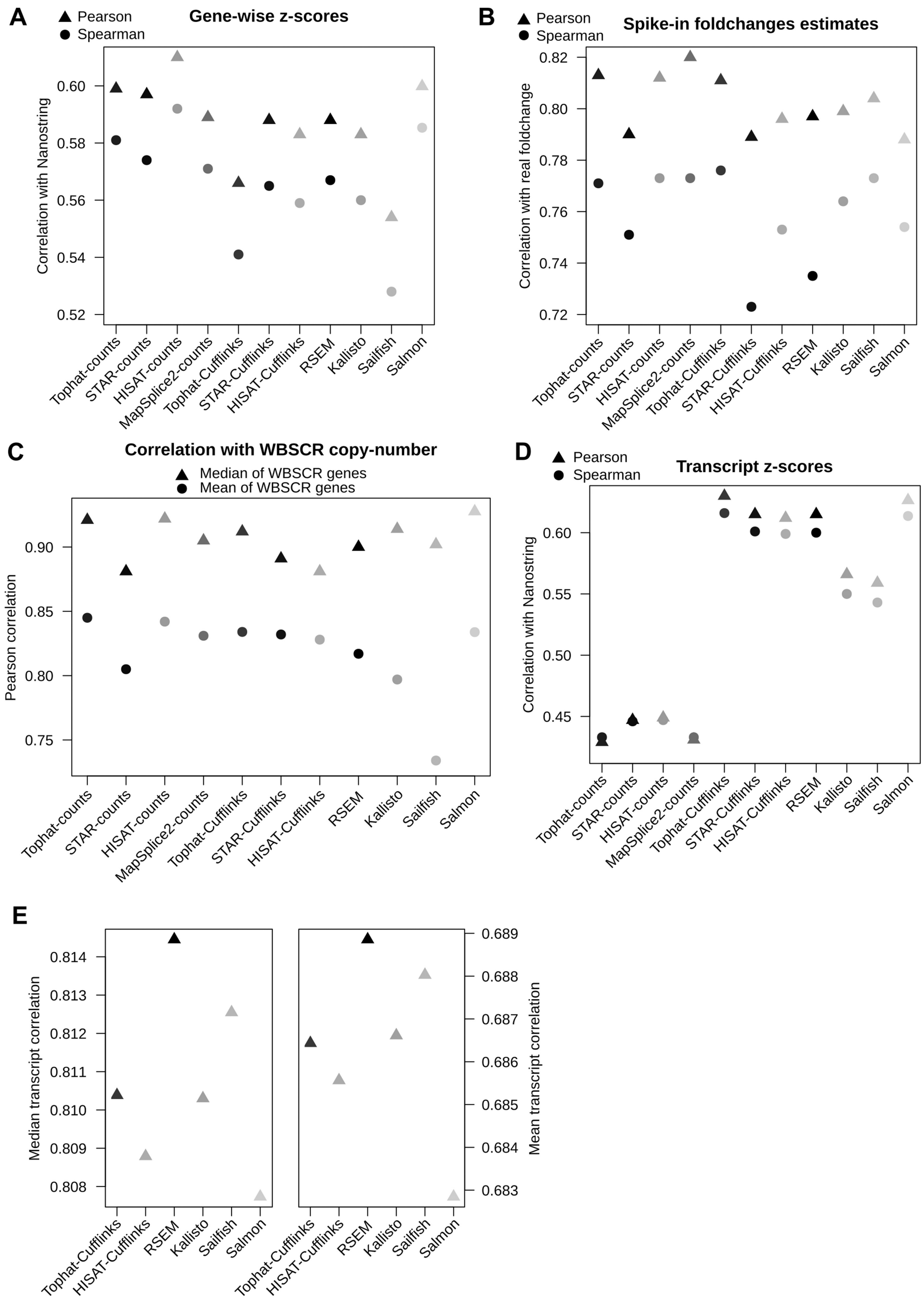


Figure 2. Accuracy of relative quantification. (**A**) Correlation of gene-wise z-scores (across samples) with Nanostring. (**B**) Correlation of observed fold-changes between spike-in mixes 1 and 2 with real fold-changes. (**C**) Correlation of WBSCR genes with their copy-number. (**D**) Correlation of transcript-wise *z*-scores with Nanostring. (**E**) Median transcript-wise correlation (across samples) between observed and expected values in the simulated dataset.

Aa syndrome and Williams-Beuren syndrome, from which we derived iPSC cell lines and profiled their transcriptomes (39). Importantly, the vast majority of these genes are expressed in iPSC and have been shown to vary linearly with copy-number both at the RNA and protein levels ((39) and data not shown). We therefore tested to what extent the expression of each of these genes correlated with its copy-number using the different methods outlined above (Figure 2C). Again, this showed a good performance of count-based methods (except STAR-based counts), along with Salmon and closely followed by Tophat-Cufflinks and Kallisto.

Transcript-level quantification is much more difficult, and consequently it revealed more substantial differences between methods (Figure 2D). Here, count-based methods performed very poorly as expected, while Salmon, RSEM, and Cufflinks (especially Tophat-Cufflinks) performed best. The fact that count-based methods performed well at the gene-level or with spike-ins, and very badly at transcript-level (which is beyond their purpose), corroborates the idea that their limitation lies in their inability to use ambiguous reads. We looked more closely at the differences between the other methods, which are very robust across a variety of metrics (see Supplementary Table S2), and concluded that the Tophat-Cufflinks pipeline is superior to others in all respects, except for estimating the absolute abundance of lowly-expressed transcripts, where RSEM performs best. However, the differences between Tophat-Cufflinks, RSEM and Salmon are very small, especially when looking at transcript *z*-scores, where they represent little more than 2% difference in correlation or median error. Importantly, Salmon was not among the best performing methods for absolute transcript quantification, but was slightly better than RSEM when looking at transcript z-scores. We therefore conclude that these three approaches appear to most accurately quantify relative differences.

Confirmation with *in-silico* data

The spike-ins had the limitation that they are not spliced and do not produce alternative isoforms from the same genomic region, while the Nanostring dataset described so far had the limitation that most probes hybridize with more than one isofrom, and that it was not assayed on the very same RNA extraction as the RNA-seq. We therefore sought to further solidify our analysis using in-silico data generated for two groups of four samples each (see methods and supplementary materials).

Of note, the simulated data is deliberately highly artificial in that the distributions of expression levels across features does not resemble a real experiment, but was rather specifically designed to make transcript quantification difficult (see methods). In particular, most isoforms of most genes were expressed, and transcripts were assigned a foldchange randomly, independently of which gene they belonged to. Therefore, while this dataset can reveal differences between methods in dealing with difficult loci, it does not show the effective relevance of those differences in a normal context (see also (14)), and should be interpreted as a further validation of the previous results.

We first calculated each transcript's foldchange between groups, and correlated them to the real ones. We were surprised to see major differences (Supplementary Figure S6A) between methods which, so far, had given comparable results, and while investigating the issue we noted that this difference was mostly due to very few transcripts that were assigned an extremely high foldchange (Supplementary Figure S6B). We therefore calculated, for each transcript, the correlation between real and measured expression across samples, and plotted the distribution of correlations (Supplementary Figure S7). The mean and median transcript correlation of the best performing methods are plotted in Figure 2E. While the differences are once again small in magnitude, RSEM performed best, followed by Sailfish. Thus, when considering both the *in silico* data and the comparison with Nanostring, RSEM appears to give the best performance.

Confirmation with a different dataset

Given the relatively small magnitude of the differences observed between the top-performing methods, we further validated them in a different, smaller dataset of 6 samples for which the very same RNA extraction was used for both RNAseq and the aforementioned Nanostring panel (Figure 3). RSEM was again the best method in quantifying transcript abundances, with a high correlation (Figure 3A) and a low median absolute error (Figure 3B) when compared to Nanostring, while Tophat-Cufflinks had the lowest mean correlation and the highest median absolute error. In terms of relative transcript quantification, Tophat-Cufflinks performed best for lowly-expressed transcripts (Figure 3E), but alignment-free methods showed the highest correlation of *z*-scores overall (Figure 3D) and among highly-expressed transcripts (Figure 3F). In addition, Salmon and RSEM were superior to other methods in estimating the ratio between different EIF4H isoforms (Figure 3C), a gene of the WBSCR for which both expressed isoforms were independently assayed by Nanostring. While these results bring nuance to the previous ones, they also corroborate the superiority of RSEM in estimating abundance, and further support Salmon for relative quantification, in agreement with the results obtained from the previous Nanostring dataset.

Difficult genes

It was recently reported that some genes are particularly difficult to quantify via RNAseq, mostly due to multi-mapping reads (11). To test whether the difficulty in estimating the abundance of these transcripts translated into a difficulty to estimate their relative expression across samples, we compared their transcript-wise correlation in our simulated data (Supplementary Figure S8). We found no significant difference in the accuracy of the quantification of transcripts from 'difficult genes' versus the rest of the transcriptome, with a 95% confidence interval of the difference in mean absolute $\log_2$(fc) deviation between 'difficult genes' and the rest of the transcriptome of -0.037 to 0.054 ($P \sim 0.71$). Once more, this corroborates the idea that it is not necessary to accurately estimate abundance, in order to accurately estimate differences across samples.

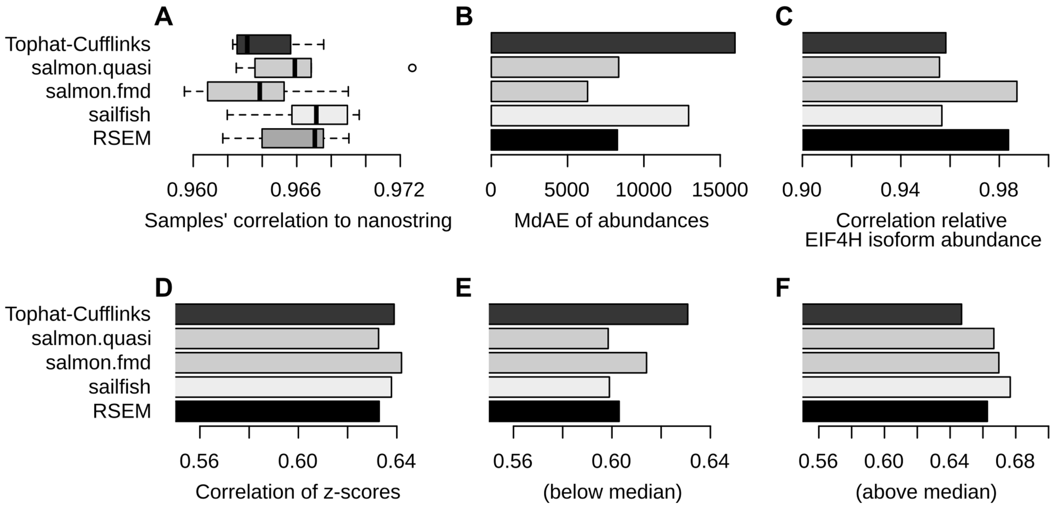


Figure 3. Accuracy of transcript quantification in the validation dataset for top-ranking methods. (**A**) Distribution of samples' Pearson correlation with Nanostring. (**B**) Median absolute error of abundance estimates. (**C**) Correlation of EIF4H isoform ratios with Nanostring. Pearson correlation of transcript-wise *z*-scores with Nanostring, for all transcripts (**D**), those expressed below the median (**E**), and those expressed above the median (**F**).

Although the relative expression of 'difficult genes' is no less accurately quantified, it is possible that differences in those features are more difficult to detect due to discarded ambiguous reads. Indeed, an analysis of variance showed that log-transformed *P*-values for differentially-expressed transcripts belonging to multi-mapping groups tended to be larger ($P \sim 2.3e{-}5$, blocking for real foldchange and read count), supporting the idea that although the relative quantification of these features is not more difficult, the identification of statistically significant differences is indeed less efficient.

Finally, to investigate whether some characteristics of the transcripts might explain the difficulty in their relative quantification, we performed an analysis of variance on the transcript-wise correlations with expected values. As expected, the most important association was with the magnitude of the foldchange ($P < 2e{-}16$), followed by the number of transcripts associated to the same gene ($P \sim 3.87e{-}06$). We also observed a weak association with transcript length ($P \sim 0.0061$), but no significant association with the number of exons ($P \sim 0.767$). For each transcript, the correlation between expected and observed quantifications across samples is reported in Supplementary Table S7.

Spike-in differential expression analysis

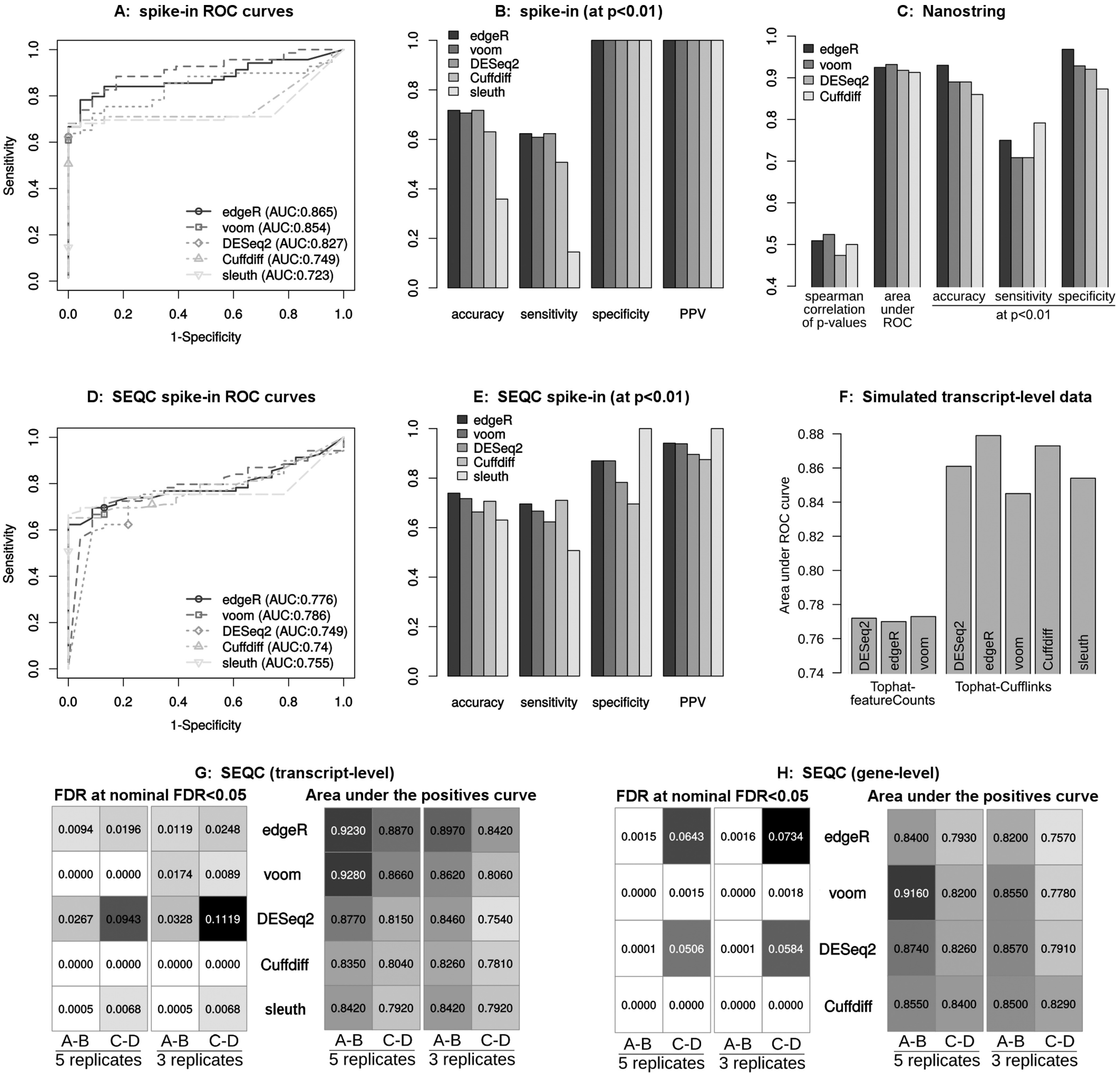
We next tested the ability of differential expression analysis (DEA) methods to accurately detect differences in concentration between the two spike-in mixes. For all methods, we calculated normalization factors in the same way, using the trimmed mean of *M*-values (TMM) method (36) on the homogenized mixes (see methods). We then computed the receiver operating characteristic (ROC) curve, which plots sensitivity against false positive rate at different significance thresholds (Figure 4A). In addition, we observed the sensitivity and specificity at different *P*-value thresholds, fold-change deviations, and the distribution of *P*-values by expected foldchange for each method (see Supplementary Figures S9–S16). As expected given the absence of alternative isoforms for spike-ins (and hence virtually no ambiguous reads), the quantification method used had very little impact on DEA accuracy. Moreover, all DEA methods appeared fairly specific, identifying no false positive at $P < 0.01$, but there were important differences in sensitivity (Figure 4A and B). The ROC curves summarize well the performance of each method (Figure 4A). EdgeR performed best, closely followed by limma/voom, while Cuffdiff and especially Sleuth were the least performing. For Sleuth, increasing the number of bootstrap samples (from 10 to 100) made no noticeable difference to its performance (Supplementary Figures S10–S11).

Upon repeating the same analysis on an external dataset (SEQC; see Materials and Methods and (9)) which also included the two spike-in mixes, the results were similar, except that Sleuth proved more specific and Cufflinks more sensitive at $P < 0.01$, although with overall performances below voom and edgeR (Figure 4D and E).

Finally, since some users are (against recommendations of the edgeR and DESeq documentations) using TPM estimates directly for DEA, we tested the performance of edgeR and voom with these values. While using TPM rather than count estimates resulted in a mild loss of specificity with voom (Supplementary Figure S17), with edgeR it resulted in a substantial loss of sensitivity (Supplementary Figure S18). Therefore, while the use of TPM values for DEA is not advisable, users who wish to do so should preferably use voom.

Differential expression analysis compared to Nanostring and simulated data

Since the spike-ins are neither spliced nor overlapping, we sought to confirm these results using the Nanostring panel (Figure 4C and Supplementary Figures S19–S22). While voom had a slightly higher area under the ROC curve and correlation with Nanostring *P*-values, edgeR again performed best overall, with the highest accuracy at $P < 0.01$. Of note, however, Cuffdiff proved to have the highest sensi-



G: SEQC (transcript-level)

	FDR at nominal FDR<0.05				Area under the positives curve			
	5 replicates		3 replicates		5 replicates		3 replicates	
	A-B	C-D	A-B	C-D	A-B	C-D	A-B	C-D
edgeR	0.0094	0.0196	0.0119	0.0248	0.9230	0.8870	0.8970	0.8420
voom	0.0000	0.0000	0.0174	0.0089	0.9280	0.8660	0.8620	0.8060
DESeq2	0.0267	0.0943	0.0328	0.1119	0.8770	0.8150	0.8460	0.7540
Cuffdiff	0.0000	0.0000	0.0000	0.0000	0.8350	0.8040	0.8260	0.7810
sleuth	0.0005	0.0068	0.0005	0.0068	0.8420	0.7920	0.8420	0.7920

H: SEQC (gene-level)

	FDR at nominal FDR<0.05				Area under the positives curve			
	5 replicates		3 replicates		5 replicates		3 replicates	
	A-B	C-D	A-B	C-D	A-B	C-D	A-B	C-D
edgeR	0.0015	0.0643	0.0016	0.0734	0.8400	0.7930	0.8200	0.7570
voom	0.0000	0.0015	0.0000	0.0018	0.9160	0.8200	0.8550	0.7780
DESeq2	0.0001	0.0506	0.0001	0.0584	0.8740	0.8260	0.8570	0.7910
Cuffdiff	0.0000	0.0000	0.0000	0.0000	0.8550	0.8400	0.8500	0.8290

Figure 4. Accuracy of differential expression analysis (DEA) methods in different contexts. (**A**) ROC curve of each method in comparing spike-in mixes (see Supplementary Figures S9 to S16 for more detail). (**B**) Accuracy metrics at $P < 0.01$ in the comparison between spike-in mixes. (**C**) Area under the ROC curve and accuracy metrics at $P < 0.01$ when comparing to Nanostring data. (**D**) ROC curves for the spike-ins of the SEQC data. (**E**) Accuracy metrics at $P < 0.01$ for the spike-ins of the SEQC data. (**F**) Area under the ROC curve of transcript-level DEA methods in the simulated data. False discovery rate (FDR) and area under the positives' curve for each method when calling differential expression within and across cell lines, at the transcript (**G**) and gene (**F**) levels (see Supplementary Figures S27 to S30 for more detail).

tivity at $P < 0.01$, despite being less accurate overall (Sleuth was not tested here because it does not perform gene-level differential expression, and the Nanostring panel had too few differentially-expressed probes hybridizing with a single transcript to perform a meaningful test at the transcript-level).

To further confirm these results and extend them to a context where isoform deconvolution is particularly challenging, we ran the same analysis methods on the simulated dataset (Figure 4F). As expected, using transcript-level quantification from featureCounts resulted in a major loss of accuracy, and we therefore focused on Cufflinks quantifications. In contrast to the spike-ins, in this context voom had the worst performance, and while edgeR gave the best performance, it was closely followed by Cuffdiff, which was superior to DESeq2 and voom.

Differential expression analysis within and across SEQC data

Given the contrasting results of Cuffdiff and Sleuth across the different datasets, we extended our tests by reanalyzing the SEQC data (9), which include transcriptomes from two human RNA samples ('A' and 'B', respectively standing for the Universal Human Reference RNA and the Human Brain Reference RNA), as well as from two mixtures of those samples ('C' and 'D') in known ratios (respectively 3:1 and 1:3), each with five replicates. In contrast to the previous cases (Nanostring, spike-ins, and simulated data), here the true differential expression is not known, and the working assumption behind this dataset is that a detected differential expression between groups (i.e. mixtures) is putatively true, while a detected differential expression within groups (i.e. between replicates of a given sample) is a false positive. Therefore, plotting the number of genes in both comparisons that are below a sliding nominal false-discovery rate (FDR) threshold gives a picture analogous to the ROC curve, which we call here a 'positives' curve' (see Supplementary Figures S23–S26). Of note, the RNA mixtures are very different from one another, and the high degree of replication makes this a rather simple task for differential expression analysis. For this reason, we recommend focusing on the (slightly more subtle) C versus D comparison.

Given that, in this context, the number of differentially-expressed genes/transcripts is unknown, we compared for each method the area under the positives' curve, and the actual false-discovery rate when using a nominal FDR threshold of 0.05 (Figure 4G–H; an equivalent of this Figure using instead unadjusted *P*-values in available in Supplementary Figure S27). We compared these values when performing the analysis at the level of transcripts (Figure 4G) or genes (Figure 4H), and when using all five or only a subset (3) of the replicates. EdgeR and voom had overall the best area under the positives' curve. Although Cuffdiff generally had a poor area under the positives' curve, it systematically had the lowest FDR (reporting no false positive), closely followed by voom and, at the transcript level, Sleuth. A possible explanation is that Cuffdiff and Sleuth use information ignored by other approaches (namely the bootstrap samples for Sleuth and the uncertainty in count estimates for Cuffdiff), which could improve their performance especially at the transcript level and with low counts or replicates. Indeed, at the transcript-level with only three replicates, the beginning of the positives' curve shows for Cuffdiff and Sleuth a greatly improved performance (see especially right panels of Supplementary Figure S30), which diminishes however upon increasing the threshold. Nevertheless, in this assay voom had the best performance, with a FDR closely matching that of Cuffdiff and Sleuth, and an area under the positives' curve on a par with edgeR (Figure 4G–H).

Given the assumption underlying the SEQC analysis that any detected differential expression between cell lines is true (9), these results must be interpreted with care. However, in light of the previous results (Figure 4C–F), they suggest that the additional information used by Cuffdiff and Sleuth might increase performance of transcript-level differential expression in some contexts, and eventually lead to the development of methods that combine the robust statistical methods of voom and edgeR with this additional information.

Effect of library type and size

To probe the impact of lower coverage on the accuracy of relative quantification, we performed a downsampling analysis using one pseudoalignment method (Salmon) and one alignment method (HISAT-Cufflinks). By and large, both methods were equally sensitive to coverage (Figure 5). Spike-in detection rate increased steadily with coverage without reaching a plateau (Figure 5A), while the transcript detection rate plateaued at around 40–50% of the reads, which correspond to roughly 25–30M pairs of reads (Figure 5C). The accuracy of relative quantification appeared instead to reach a plateau already at 30% of reads, or at about 19M reads (Figure 5B–D).

We next assessed the effect of library type and read size on the accuracy of the quantification. To emulate single-end reads, we aligned and analysed only the first mates of each read pair, while to emulate shorter reads, we trimmed the right-most 50 bp of all original reads. We then compared this derivative quantification with that obtained with the original reads. For transcript-level abundance estimates, 50 bp single-end reads proved highly inaccurate, with a very low correlation to Nanostring (Figure 5E) and a very high median absolute error (Figure 5F). A major part of this massive drop in correlation (0.4 compared to 0.9 and above for other library types) appears to be due to few undetected or under-estimated transcripts, especially for Salmon (Supplementary Figures S28–S29). Nevertheless, the observation of the same pattern in median absolute error (where Salmon instead shows lower error than Cufflinks, see Figure 5F) suggests that the effect cannot entirely be attributed to those few transcripts. These results indicate that either long (100 bp) reads or paired reads are needed, with paired reads providing a slightly better improvement (Figure 5E and F). Of note, the respective improvements provided by longer or paired reads were not additive, but rather redundant. A similar pattern could be observed on the accuracy of relative transcript quantification (Figure 5G), with paired-reads again providing the best relative quantification, although differences in this case were much smaller. However, when we measured the capacity of the different libraries to detect the ratio between the two expressed isoforms of EIF4H (independently measured by two Nanostring probes), it was instead read size that had the strongest impact (Figure 5H). Together, these results suggest that either long (100 bp) or paired reads are very important for transcript-level quantification, but that they offer improvements that are partially redundant with each other.

Finally, we assessed the extent to which library size affected each DEA method (Figure 5I). Since the downsampling analysis had suggested that 30% of the reads were sufficient for accurate quantification, we tested differential expression analysis when using 100%, 30% and 10% of the reads. DESeq2 and edgeR were most affected by the loss in coverage, closely followed by Cuffdiff. Sleuth and voom were less affected, with Sleuth showing very little reduction of accuracy with 30% of the reads, and only dropping when using 10%.

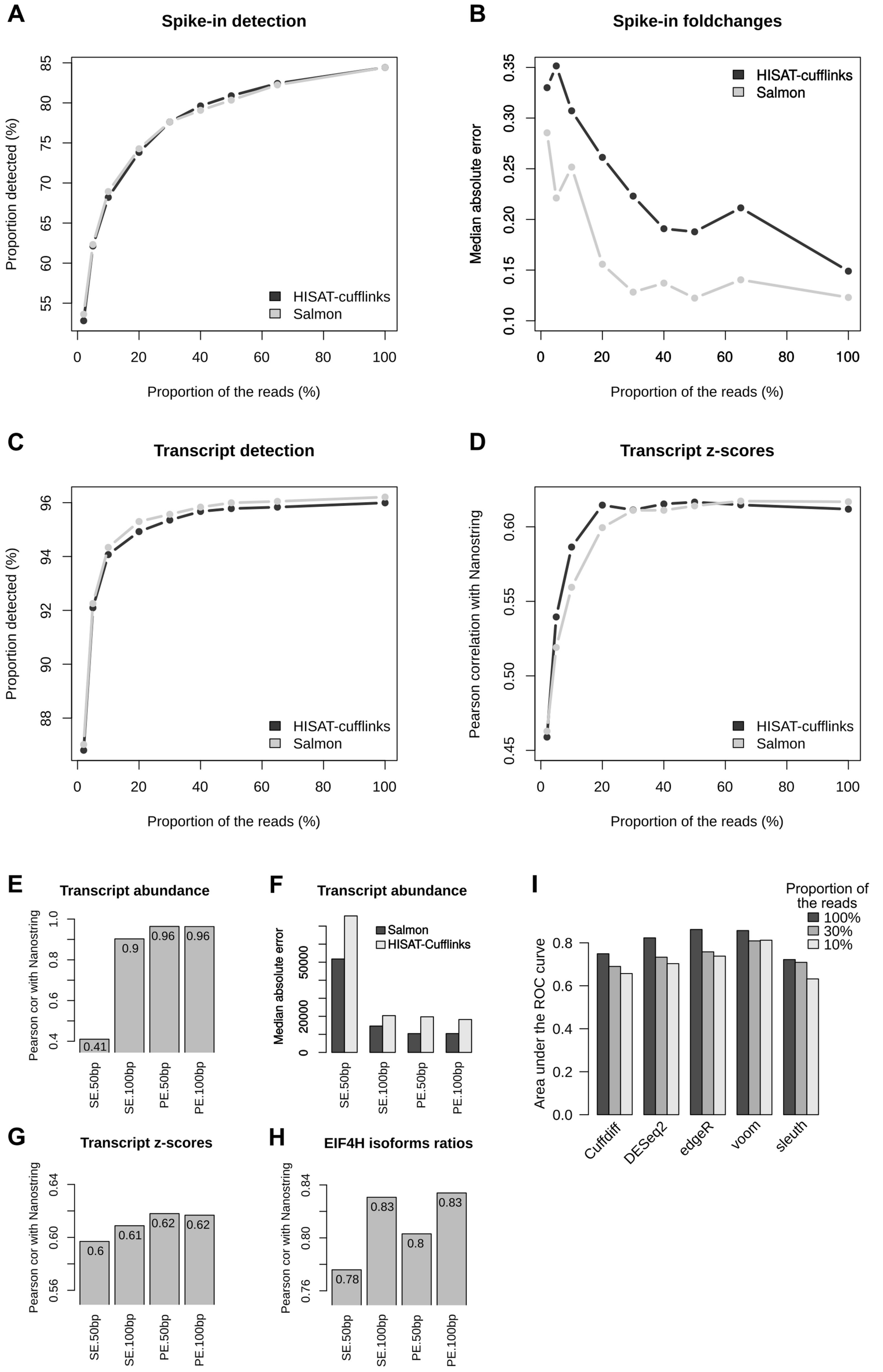


Figure 5. Downsampling analyses. Effect of downsampling coverage (i.e. library size) on spike-in detection rate (**A**), median absolute error in spike-in foldchange estimations (**B**), transcript detection rate (**C**), and transcript-wise z-score estimation (**D**), using both an alignment based method (HISAT-Cufflinks) and a pseudo-alignment method (Salmon). **E-H:** Effect of library type and read length on the correlation with Nanostring, specifically for transcript abundance estimates (**E** and **F**), transcript z-score estimates (**G**), and estimates of the ratio between EIF4H isoforms (**H**). (**I**) Effect of coverage on the detection of differences between the spike-in mixes.

Time and memory requirements

Finally, in order to assess the computing resources required by each software, we measured the actual real time of computation (wall clock time using eight cores) and the maximum peak of RAM memory used by each software for processing each sample (Supplementary Figure S30). A first major distinction must be made between alignment-based and pseudo-alignment based pipelines, with the former requiring considerably more time. Among the aligners, the best results were obtained by HISAT and STAR, taking ~1 h per sample, while Tophat required 9 times as long. For quantification, Cuffquant required ~7 h per sample, while featureCounts's average time was less than 45 minutes per sample. RSEM, which performs both alignment and quantification, performed better than any pipeline involving Cuffquant. All pseudo-alignment software outperformed alignment-based methods, and in particular Sailfish was the fastest with around 7 minutes per sample, roughly half of the time required by Kallisto and Salmon (disabling bootstraps). Finally, Salmon was considerably faster than Kallisto at generating bootstraps samples, merely scaling linearly while Kallisto took over a day per sample with 10 bootstraps.

Memory consumption is a major problem for STAR, which required over 32GB of RAM. All the other aligners required a reasonable amount of memory, with HISAT performing slightly better than Tophat. Cuffquant had a peak consumption of almost 7GB, while featureCounts had a very low memory footprint. Among pseudo-aligners, Sailfish was the most demanding with a peak memory requirement of 5GB.

Given that differential expression software ran relatively fast, we did not compile or compare their processing time and requirements.

DISCUSSION

Relative (i.e. across-samples) quantification is critical for most RNAseq applications, especially given the increasing prominence of transcriptomics in the unravelling of human diseases and the attending need to capture meaningful relative differences within large cohorts of often highly heterogeneous samples. Here, we have shown that the methods providing the best absolute quantification (i.e. across-genes within individual samples) are not necessarily those that provide good relative quantification. Consequently, benchmarking should pay more attention to relative differences across samples, focusing on gene- and transcript-wise foldchanges and *z*-scores, as we have done here, or using other approaches (40).

This approach allowed us to note that although methods based on counting reads/fragments unambiguously overlapping a feature systematically give worse absolute quantifications than alternatives, they provide equal or even superior results for assessing relative expression at the gene-level, warranting their use for gene-level differential expression analysis. At the transcript-level, Cufflinks, RSEM and Salmon had a comparable performance, with RSEM being superior for absolute (but not relative) quantification. It is particularly noteworthy that Salmon, which (like Sailfish and Kallisto) bypasses traditional alignment and thereby quantifies a single sample in a matter of minutes, had a comparable performance to Cufflinks and RSEM. Importantly, we confirmed these results using a variety of assays on both empirical and simulated data.

We next benchmarked differential expression analysis methods. Most popular methods gave fairly good results, and while relevant differences could be observed, all benchmarks did not uniformly agree, which again shows the necessity of a plurality of assays to compare these methods. Nevertheless, we can say that in general voom and edgeR showed the most stable performance, being superior to alternatives in most assays, with voom significantly underperforming only in the (highly challenging) transcript-level simulation (Figure 4F), and edgeR showing suboptimal results only in the SEQC dataset (Figure 4G-H). In a few of the tests (especially at lower coverage or sample size), Sleuth and Cuffdiff appeared promising for transcript-level analysis, but overall they proved inferior to alternative methods. This suggests that further development might allow to harness the additional information used by Sleuth and Cuffdiff within a more robust statistical framework.

Finally, we studied the effect of library size and type on the accuracy of RNAseq quantification, and showed that either long (100 bp) reads or (slightly better) paired reads were needed for transcript-level quantification, although the respective improvements provided are partially redundant with each other. Indeed, using single-end 50 bp reads resulted in a massive drop in correlation with Nanostring and a corresponding increase in median absolute error (Figure 5E and F). Instead, coverage had little impact on quantification above a certain minimum (roughly 20M reads with this experimental design), a relatively small impact on DEA, but a considerable impact on the capacity to detect spike-ins, which had not yet plateaued at full library size. This suggests that although high coverage might improve detection and quantification of rare transcripts, it is not particularly useful for the quantification of most of the transcriptome. Obviously, coverage might still be useful for other purposes, such as transcript assembly, study of RNA editing, etc., but in general it seems much preferable to invest in paired libraries and/or longer reads.

Most packages include a variety of options, whose multiple combinations clearly escape the range of tests we could perform. In addition, new methods are continually being released. For this reason, we created a R package, as well as an online platform, enabling researchers to apply the same extensive benchmarking we have performed to their pipeline of choice by simply uploading the quantification. The platform produces and displays above 30 diagnostic plots for each pipeline uploaded, and allows the comparison of a large array of accuracy measurements (including correlations and median absolute errors of values, foldchanges and feature-wise z-scores) across methods, thereby providing a rich resource for continued benchmarking efforts.

By design this benchmarking resource cannot settle all aspects of RNAseq analysis. In particular, it cannot assess normalization methods because the Nanostring quantification relies on housekeeping genes for normalization; similarly, it cannot test the impact of different transcriptome annotations or assembly methods, because the Nanostring panel was built and defined on the basis of Refseq transcripts. Further efforts, and different resources, will therefore be required to address these issues. One such resource was published during the revision of this manuscript (40), featuring a complementary benchmarking platform for relative RNAseq quantification, based on standard deviation for assessing accuracy of relative expression estimates. The method was used to compare popular quantification pipelines, yielding conclusions consistent with our results, such as a comparatively small effect of the alignment method, and a very good performance of RSEM and Salmon. Of note, however, their analysis rests on very different empirical evidence, with a first dataset composed of few (4) samples probed by microarrays to detect real biological differences, and a second based on simulated differential-expression added to real transcriptomes. Moreover, their package is explicitly not designed to assess DEA methods. Therefore, the resource we offer here is comparatively broader, richer in empirical data, and based on a larger set of tests and metrics. Nevertheless, as each benchmark carries its own limitations, a careful assessment of RNAseq analyses methods benefits from a plurality of benchmarking resources.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENT

We wish to thank Alessandro Ogier for his precious help in troubleshooting software compilation and server settings.
Authors' contributions: PLG designed the benchmark and wrote the code. PLG, AV and GT wrote the paper. PLG and AV performed the analyses and created the figures. AA performed the selection and the primary analysis of the Nanostring data. PLG, AV, PL and VD ran the different pipelines. VD generated the simulated dataset. PLG and GT conceived and supervised the project.

FUNDING

European Research Council [616441 – DISEASEAVATARS to G.T.]; Regione Lombardia (Ricerca Indipendente 2012); Italian Ministry of Health (Ricerca Corrente to G.T.); ERA-NET Neuron Program (to G.T. and P.L.G.); Italian Association for Cancer Research (to G.T.); EPIGEN Flagship Project of the Italian National Research Council (to G.T.); Jerome-Lejeune Foundation (to G.T.); Umberto Veronesi Foundation (fellowship to P.L.G.); Federation of European Biochemical Societies (to A.A.); Italian Foundation for Cancer Research (to P.L. and V.D.). Funding for open access charge: ERC Research Grant DISEASEAVATARS [616441].
Conflict of interest statement. None declared.

REFERENCES

1. Engström,P.G., Steijger,T., Sipos,B., Grant,G.R., Kahles,A., Rätsch,G., Goldman,N., Hubbard,T., Harrow,J., Guigó,R. *et al.* (2013) Systematic evaluation of spliced alignment programs for RNA-seq data. *Nat. Methods*, **10**, 1185–1191.
2. Seyednasrollah,F., Laiho,A. and Elo,L.L. (2013) Comparison of software packages for detecting differential expression in RNA-seq studies. *Brief. Bioinformatics*, **16**, 1–12.
3. Soneson,C. and Delorenzi,M. (2013) A comparison of methods for differential expression analysis of RNA-seq data. *BMC Bioinformatics*, **14**, 91.
4. Kanitz,A., Gypas,F., Gruber,A.J., Gruber,A.R., Martin,G. and Zavolan,M. (2015) Comparative assessment of methods for the computational inference of transcript isoform abundance from RNA-seq data. *Genome Biol.*, **16**, 1–26.
5. Pertea,M., Pertea,G.M., Antonescu,C.M., Chang,T.-C., Mendell,J.T. and Salzberg,S.L. (2015) StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nat. Biotechnol.*, **33**, 290–295.
6. Rapaport,F., Khanin,R., Liang,Y., Pirun,M., Krek,A., Zumbo,P., Mason,C., Socci,N. and Betel,D. (2013) Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. *Genome Biol.*, **14**, R95.
7. Rehrauer,H., Opitz,L. and Tan,G. (2013) Blind spots of quantitative RNA-seq: the limits for assessing abundance, differential expression, and isoform switching. *BMC Bioinformatics*, **14**, 1–10.
8. Munro,S., Lund,S., Pine,P., Binder,H., Clevert,D.-A., Conesa,A., Dopazo,J., Fasold,M., Hochreiter,S., Hong,H. *et al.* (2014) Assessing technical performance in differential gene expression experiments with external spike-in RNA control ratio mixtures. *Nat. Commun.*, **5**, 5125.
9. Su,Z., Łabaj,P., Li,S., Thierry-Mieg,J., Thierry-Mieg,D., Shi,W., Wang,C., Schroth,G., Setterquist,R., Thompson,J. *et al.* (2014) A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium. *Nat. Biotechnol.*, **32**, 903–914.
10. Rajkumar,A.P., Qvist,P., Lazarus,R., Lescai,F., Ju,J., Nyegaard,M., Mors,O., Børglum,A.D., Li,Q. and Christensen,J.H. (2015) Experimental validation of methods for differential gene expression analysis and sample pooling in RNA-seq. *BMC Genomics*, **16**, 548.
11. Robert,C. and Watson,M. (2015) Errors in RNA-Seq quantification affect genes of relevance to human disease. *Genome Biol.*, **16**, 177.
12. Zhao,S. and Zhang,B. (2015) A comprehensive evaluation of ensembl, RefSeq, and UCSC annotations in the context of RNA-seq read mapping and gene quantification. *BMC Genomics*, **16**, 1–14.
13. Trapnell,C., Hendrickson,D.G., Sauvageau,M., Goff,L., Rinn,J.L. and Pachter,L. (2012) Differential analysis of gene regulation at transcript resolution with RNA-seq. *Nat. Biotechnol.*, **31**, 46–53.
14. Soneson,C., Love,M.I. and Robinson,M.D. (2016) Differential analyses for RNA-seq: transcript-level estimates improve gene-level inferences. *F1000Research*, **4**, 1521.
15. Zhou,X. and Robinson,M.D. (2015) Do count-based differential expression methods perform poorly when genes are expressed in only one condition? *Genome Biol.*, **16**, 222.
16. Soneson,C., Matthes,K.L., Nowicka,M., Law,C.W. and Robinson,M.D. (2016) Isoform pre-filtering improves performance of count-based methods. *Genome Biol.*, **17**, 1–15.
17. Patro,R., Mount,S.M. and Kingsford,C. (2014) Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms. *Nat. Biotechnol.*, **32**, 462–464.
18. Wang,Z., Gerstein,M. and Snyder,M. (2009) RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, **10**, 57–63.
19. Wang,C., Gong,B., Bushel,P.R., Thierry-Mieg,J., Thierry-Mieg,D., Xu,J. and Tong,W. (2014) The concordance between RNA-seq and microarray data depends on chemical treatment and transcript abundance. *Nat. Biotechnol.*, **32**, 926–932.
20. Tong,W., Lucas,A., Shippy,R., Fan,X., Fang,H., Hong,H., Orr,M., Chu,T., Guo,X., Collins,P. *et al.* (2006) Evaluation of external RNA controls for the assessment of microarray performance. *Nat. Biotechnol.*, **24**, 1132–1139.
21. Geiss,G., Bumgarner,R., Birditt,Brian, Dahl,T., Dowidar,N., Dunaway,D., Fell,H., Ferree,S., George,R., Grogan,T. *et al.* (2008) Direct multiplexed measurement of gene expression with color-coded probe pairs. *Nat. Biotechnol.*, **26**, 317–325.
22. Trapnell,C., Roberts,A., Goff,L., Pertea,G., Kim,D., Kelley,D.R., Pimentel,H., Salzberg,S., Rinn,J. and Pachter,L. (2012) Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat. Protoc.*, **7**, 562–578.

23. Dobin,A., Davis,C.A., Schlesinger,F., Drenkow,J., Zaleski,C., Jha,S., Batut,P., Chaisson,M. and Gingeras,T.R. (2012) STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, **29**, 15–21.
24. Kim,D., Langmead,B. and Salzberg,S.L. (2015) HISAT: a fast spliced aligner with low memory requirements. *Nat. Methods*, **12**, 357–360.
25. Wang,K., Singh,D., Zeng,Z., Coleman,S.J., Huang,Y., Savich,G.L., He,X., Mieczkowski,P., Grimm,S., Perou,C. *et al.* (2010) MapSplice: accurate mapping of RNA-seq reads for splice junction discovery. *Nucleic Acids Res.*, **38**, e178.
26. Anders,S., Pyl,P.T. and Huber,W. (2015) Genome analysis HTSeq — a Python framework to work with high-throughput sequencing data. *Bioinformatics*, **31**, 166–169.
27. Liao,Y., Smyth,G.K. and Shi,W. (2013) featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics*, **30**, 923–930.
28. Li,B., Ruotti,V., Stewart,R.M., Thomson,J.A. and Dewey,C.N. (2009) RNA-Seq gene expression estimation with read mapping uncertainty. *Bioinformatics*, **26**, 493–500.
29. Langmead,B., Trapnell,C., Pop,M. and Salzberg,S.L. (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.*, **10**, R25.
30. Anders,S. and Huber,W. (2010) Differential expression analysis for sequence count data. *Genome Biol.*, **11**, R106.
31. Love,M.I., Huber,W. and Anders,S. (2014) Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol.*, **15**, 1–34.
32. Law,C.W., Chen,Y., Shi,W. and Smyth,G.K. (2014) voom: Precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol.*, **15**, R29.
33. Ritchie,M.E., Phipson,B., Wu,D., Hu,Y., Law,C.W., Shi,W. and Smyth,G.K. (2015) limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.*, **43**, e47.
34. Robinson,M.D., McCarthy,D.J. and Smyth,G.K. (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140.
35. Irizarry,R.A., Wu,Z. and Jaffee,H.A. (2006) Comparison of Affymetrix GeneChip expression measures. *Bioinformatics*, **22**, 789–794.
36. Robinson,M.D. and Oshlack,A. (2010) A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology*, **11**, R25.
37. Frazee,A.C., Jaffe,A.E., Langmead,B. and Leek,J.T. (2015) Polyester: simulating RNA-seq datasets with differential transcript expression. *Bioinformatics*, **31**, 2778–2784.
38. Li,H., Handsaker,B., Wysoker,A., Fennell,T., Ruan,J., Homer,N., Marth,G., Abecasis,G., Durbin,R. and 1000 Genome Project Data Processing Subgroup. (2009) The sequence Alignment/Map format and SAMtools. *Bioinformatics*, **25**, 2078–2079.
39. Adamo,A., Atashpaz,S., Germain,P.-L., Zanella,M., D'Agostino,G., Albertin,V., Chenoweth,J., Micale,L., Fusco,C., Unger,C. *et al.* (2015) 7q11.23 dosage-dependent dysregulation in human pluripotent stem cells affects transcriptional programs in disease-relevant lineages. *Nat. Genet.*, **47**, 132–141.
40. Teng,M., Love,M.I., Davis,C.A., Djebali,S., Dobin,A., Graveley,B.R. *et al.* (2016) A benchmark for RNA-seq quantification pipelines. *Genome Biol.*, **17**, 74.